An Algorithm for Generation of Efficient Manipulator Dynamic Equations

Joel W. Burdick
Artificial Intelligence Laboratory
Stanford University

Abstract

This paper presents a method for the generation of efficient manipulator dynamic equations in symbolic form. The efficiency is obtained by the use of simplification rules during the process of equation generation. These simplifications are based on the structure of manipulator dynamics, on simplifications that arise from common manipulator geometries, and from other heuristic simplification rules. This algorithm has been implemented in a lisp-based program, EMDEG (Efficient Manipulator Dynamic Equation Generator). The development of the algorithm, the derivation of simplification rules, and some implementation aspects are discussed; and an example is presented.

1. Introduction

The topic of manipulator dynamics has received considerable attention from the robotics research community. Various formulations, such as Lagrangian [Paul 1981], recursive Lagrangian [Hollerbach 1980], recursive Newton-Euler [Luh, Walker, Paul], and Kane's method [Kane] have been proposed for the formulation of the manipulator dynamics equations.

One of the primary concerns in manipulator dynamics is computational efficiency. In the numerical procedures, efficiency is gained by clever reorganization of the computations. However, analytical approaches to the generation of manipulator equations can be more efficient than numerical approaches for a given manipulator [Kane][Renaud], and are often necessary for some problems in manipulator analysis, design [Khatib and Burdick], and control. Unfortunately, for typical manipulators the generation of these equations by hand is at best a lengthy and tedious process. To overcome this problem, several programs [Cesareo][Murray] have been developed to generate the equations of motion in symbolic form. However, these programs have been based on inefficient Lagrangian formulations, and produce expressions which are not as computationally efficient as the recursive numerical procedures.

In this paper, a method for generating manipulator equations of motion in a symbolic and compact form is developed, and some simplification rules based on both the structure of serial chain dynamic equations and on common manipulator geometries are presented. These rules form the basis for a lisp-based program, EMDEG (Efficient Manipulator Dynamic Equation Generator), to automatically generate manipulator dynamic equations.

2. Geometric/Kinematic Conventions

In the derivation of this method, it is assumed that a manipulator consists of a serial chain of N rigid bodies connected by uniaxial joints: prismatic or revolute. More complex joints can be modeled by adding "phantom" bodies which have no mass, inertia, or size. The base of the manipulator is fixed in an inertial reference frame, and all external forces, except for gravity and actuators effects, are neglected. E.g., this model does not deal with forces applied to the end-effector.

The kinematic convention presented here follows the modified Denavit-Hartenberg notation and conventions as developed in [Craig]. The kinematic relationship between two bodies in a serial chain mechanism connected by uniaxial joints can be described by a set of parameters $\{\alpha, d, a, \theta\}$. The definitions of these parameters (see figure 1) are as follows:

1. For the i^{th} link, establish a coordinate system, $\mathcal{C}_i$, which is represented by the orthogonal unit vectors $\mathbf{x}_i, \mathbf{y}_i$, and $\mathbf{z}_i$. A coordinate system $\mathcal{C}_i$ is different from a frame $\mathcal{F}_i$ in that a frame has its origin fixed to some location (such as a point on a link), whereas a coordinate system is defined independently of location. The unit vectors of $\mathcal{F}_i$ and $\mathcal{C}_i$ are equivalent.
2. Attach $\mathcal{F}_i$ to the i^{th} link such that $\mathbf{z}_i$ is aligned with the joint axis.
3. The twist angle, α_{i-1}, is defined as the angle between the projection of $\mathbf{z}_i$ and $\mathbf{z}_{i-1}$.
4. The length parameter, a_{i-1} is defined as the mutually orthogonal distance between $\mathbf{z}_i$ and $\mathbf{z}_{i-1}$.
5. d_i is the distance along $\mathbf{z}_i$ between the intersections of a_{i-1} and a_i with $\mathbf{z}_i$.
6. The angle θ_i is defined as the angle between the projections of a_{i-1} and a_i

Using the above notation, vectors described in $\mathcal{C}_i$ can be transformed to a description in $\mathcal{C}_{i-1}$ by the 3×3 rotation matrix, ${}^{i-1}\mathbf{R}^i$, and the vector from the origin of $\mathcal{F}_{i-1}$ to the origin of $\mathcal{F}_i$ is denoted by $\mathbf{P}_i$:

$$
\begin{aligned}
{}^{i-1}\mathbf{R}^i &= \begin{pmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} \end{pmatrix} \\
\mathbf{P}_i &= \begin{pmatrix} a_{i-1} \\ -s\alpha_{i-1} d_i \\ c\alpha_{i-1} d_i \end{pmatrix};
\end{aligned}
\tag{1}
$$

where $c\theta_i$ denotes the cosine of θ_i, $c\alpha_i$ denotes the cosine of the twist angle α_i, etc..

3. Dynamic Equations

The following derivations are done in "joint space". That is, the generalized coordinates $\mathbf{q}$ are the joint angles θ for revolute joints, or the distance d for prismatic joints, and the generalized speeds are the time derivatives of the generalized coordinates. The active forces consist solely of actuator torques and forces and gravity.

In the following derivations, the superscript "$*$" ("o") on vectors, matrices, and dyads denotes that these quantities are described with respect to the center of mass (origin of a frame). For vectors, post superscripts index the link number, and leading superscripts indicate that the vector is described in a particular coordinate system or frame. For example the velocity vector ${}^jV^{i*}$ is the velocity of the center of mass of link i, as described in reference coordinate system (or reference frame) $\mathcal{C}_j$ ($\mathcal{F}_j$).

Again, let ${}^j\bar{V}^{i^*}$ denote the velocity of the center of mass of link i as expressed in C_j; and let ${}^j\bar{w}^i$ denote the angular velocity of link i as expressed in C_j. If we use Kane's concept of partial velocities [Kane], then ${}^j\bar{V}^{i^*}$ and ${}^j\bar{w}^i$ can be expanded in terms of ***partial velocities*** and ***partial angular velocities***:

$$ {}^j\bar{V}^{i^*} = \sum_{k=1}^{N} {}^j\tilde{\bar{V}}^i_k \dot{q}_k \qquad {}^j\bar{w}^i = \sum_{k=1}^{N} {}^j\tilde{\bar{w}}^i_k \dot{q}_k \tag{2} $$

where ${}^j\tilde{\bar{V}}^i_k$ is termed a "partial velocity", and is the partial derivative of the velocity of link i with respect to generalized speed $\dot{q}_k$, as expressed in C_j :

$$ {}^j\tilde{\bar{V}}^i_k = \frac{\partial\, {}^j\bar{V}^{i^*}}{\partial \dot{q}_k} \qquad {}^j\tilde{\bar{w}}^i_k = \frac{\partial\, {}^j\bar{w}^i}{\partial \dot{q}_k}. \tag{3} $$

For notational simplicity, the dependence of ${}^j\tilde{\bar{V}}^i_k$ and ${}^j\tilde{\bar{w}}^i_k$ on $\mathbf{q}$ is not made explicit.

The actual expressions for the partial velocities and partial angular velocities can be easily derived from the recursive nature of velocities in serial chain mechanisms. The partial angular velocity of link i with respect to joint j is:

$$ {}^i\tilde{\bar{w}}^i_j = \begin{pmatrix} {}^i\mathbf{z}_j \\ 0 \end{pmatrix} = \begin{pmatrix} {}^i\mathbf{R}^j\mathbf{z} \\ 0 \end{pmatrix} \text{ for joint j } = \begin{pmatrix} \text{revolute} \\ \text{prismatic} \end{pmatrix}. \tag{4} $$

Similarly the partial velocity of link i with respect to joint j can be derived as:

$$ {}^j\tilde{\bar{V}}^i_j = \begin{pmatrix} \hat{\mathbf{z}} \times {}^j\mathbf{L}^*_{j,i} \\ \mathbf{z} \end{pmatrix} \text{ for j } = \begin{pmatrix} \text{revolute} \\ \text{prismatic} \end{pmatrix} \tag{5} $$

where ${}^j\mathbf{L}^*_{j,i}$ is the vector from the origin of $\mathcal{F}_j$ to the center of mass of link i as expressed in C_j. ${}^j\mathbf{L}^*_{j,i}$ can be expressed as:

$$ {}^j\mathbf{L}^*_{j,i} = \left(\sum_{k=j}^{i-1} {}^j\mathbf{R}^k\mathbf{P}_{k+1}\right) + {}^j\mathbf{R}^i\mathbf{r}^*_i \tag{6} $$

where $\mathbf{r}^*_i$ is the vector from origin of $\mathcal{F}_i$ to the center of gravity of link i. It should be noted that because of the serial chain nature of manipulators:

$$ \tilde{\bar{V}}^{i^*}_j = \tilde{\bar{w}}^i_j = 0 \qquad \text{for } j > i. \tag{7} $$

Let m_i and $I^{i^*/i}$ denote the mass of link i, and inertia dyadic of link i as expressed in a coordinate system parallel to $\mathcal{F}_i$, but whose origin is located at the center of mass of link i. The kinetic energy of the manipulator can be expressed as the sum of the translational and rotational kinetic energy for each link:

$$ \begin{aligned} T = \sum_{i=1}^{N} T_i &= \sum_{i=1}^{N} \frac{1}{2}\left[m_i(\bar{V}^{i^*}\cdot\bar{V}^{i^*}) + \bar{w}^i \cdot I^{i^*/i}\cdot\bar{w}^i\right] \\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\left[m_i(\tilde{\bar{V}}^{i^*}_j\cdot\tilde{\bar{V}}^{i^*}_k) + \tilde{\bar{w}}^i_j\cdot I^{i^*/i}\cdot\tilde{\bar{w}}^i_k\right]\dot{q}_j\dot{q}_k \\ &= \frac{1}{2}\,\dot{\mathbf{q}}^T\mathbf{M}(\mathbf{q})\dot{\mathbf{q}} \end{aligned} \tag{8} $$

where $\mathbf{M}(\mathbf{q})$ is the kinetic energy, or inertia, matrix of the manipulator. By rearranging the summation above, and using (7), each element of $\mathbf{M}(\mathbf{q})$ will have the form:

$$ M_{i,j} = \sum_{k=max(i,j)}^{N} m_k(\tilde{\bar{V}}^{k^*}_i\cdot\tilde{\bar{V}}^{k^*}_j) + \tilde{\bar{w}}^k_i\cdot I^{k^*/k}\cdot\tilde{\bar{w}}^k_j. \tag{9} $$

Since $\mathbf{M}(\mathbf{q})$ is symmetric, only the upper half of the matrix need be computed, and so it will generally be assumed that $j \geq i$ for $M_{i,j}$.

The potential energy of each link will simply be $m_i g h_i$, where h_i is the reference height of each link, which can be computed as $\mathbf{z}\cdot{}^0\mathbf{L}^*_{0,i}$. The total potential energy of the system will thus be:

$$ V(\mathbf{q}) = g\sum_{i=1}^{N} m_i\left(\mathbf{z}_0\cdot{}^0\mathbf{L}^*_{0,i}\right) \tag{10} $$

If we form the Lagrangian, $L = T - V$, the equations of motion can be derived by substituting L into the Euler-Lagrange equations:

$$ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} = \mathbf{F} \tag{11} $$

to create the dynamic equations with the form

$$ \mathbf{M}(\mathbf{q}) + \mathbf{c}(\mathbf{q},\dot{\mathbf{q}}^2) + \mathbf{b}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{F} \tag{12} $$

where the elements of $\mathbf{M}(\mathbf{q})$ are given by (9); $\mathbf{c}(\mathbf{q},\dot{\mathbf{q}}^2)$, $\mathbf{b}(\mathbf{q},\dot{\mathbf{q}})$, and $\mathbf{g}(\mathbf{q})$ are the vectors of coriolis, centrifugal, and gravity forces; and $\mathbf{F}$ is the vector of joint actuator torques or forces. The i^{th} elements of $\mathbf{c}(\mathbf{q},\dot{\mathbf{q}}^2)$ and $\mathbf{b}(\mathbf{q},\dot{\mathbf{q}})$ can be expanded as:

$$ c_i(\mathbf{q},\dot{\mathbf{q}}^2) = \sum_{j=1}^{N}\mathbf{c}_{i,j,j}\dot{q}_j^2; \qquad b_i(\mathbf{q},\dot{\mathbf{q}}) = \sum_{j=1}^{N-1}\sum_{k=j+1}^{N}\mathbf{b}_{i,j,k}\dot{q}_j\dot{q}_k. \tag{13} $$

where $\mathbf{c}_{i,j,j}$ and $\mathbf{b}_{i,j,k}$ are the configuration dependent centripetal and coriolis parameters. Both $\mathbf{c}_{i,j,j}$ and $\mathbf{b}_{i,j,k}$ can be expressed in terms of the Christoffel symbol, β_{ijk}:

$$ \mathbf{c}_{i,j,j} = \beta_{ijj}; \qquad \mathbf{b}_{i,j,k} = 2\beta_{ijk} \tag{14} $$

where the Christoffel symbol has the form:

$$ \beta_{ijk} = \frac{1}{2}\left(\frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{jk}}{\partial q_i}\right). \tag{15} $$

$\mathbf{c}_{i,j,j}$ and $\mathbf{b}_{i,j,k}$ could be expressed in terms of masses, inertia dyadics, and simple vector operations on partial velocities and partial angular velocities. However, as will be shown, the expression of these parameters in terms of the Christoffel symbol is more useful for the symbolic generation of $\mathbf{c}_{i,j,j}$ and $\mathbf{b}_{i,j,k}$, since this form can lead to substantial simplifications.

The contribution of potential energy(10) in equation (11) will generate the vector of joint gravity forces, $\mathbf{g}(\mathbf{q})$. The i^{th} component of $\mathbf{g}(\mathbf{q})$ can be found as:

$$ g_i(\mathbf{q}) = -g\sum_{k=i}^{N} m_k\left({}^0\tilde{\bar{V}}^k_i\cdot\mathbf{z}\right) \tag{16} $$

4. Simplification Rules

The simplification of the dynamics equations can be categorized into the following basic areas: simplifications based on the underlying structure of the equations; simplification based on the regularity of common manipulator configurations; simplification based on some heuristic rules for simplification of dynamic equations; and general simplification procedures that apply to all algebraic systems. In this case the term simplification implies the reduction of the amount of real time computation required to compute the complete set of configuration dependent dynamic parameters, $\mathbf{M}(\mathbf{q})$,$\mathbf{b}(\mathbf{q},\dot{\mathbf{q}})$,$\mathbf{c}(\mathbf{q},\dot{\mathbf{q}}^2)$, and $\mathbf{g}(\mathbf{q})$; and subsequently, to minimize the computation of the ***inverse*** dynamic problem: the computation of $\mathbf{F}$ in equation (12) given $\mathbf{q}$,$\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$.

There are two basic ideas behind simplification:

1. Develop a hierarchy of symbols (a partial ordering) to improve computational efficiencies. In essence, factorize the equations of motion, using a set of rules to guide the factorization and simplification.

2. Whenever possible, segregate the computations into configuration dependent and configuration independent portions, since configuration independent parameters can be computed once, and then stored as constants.

A. Simplification of Inertia Terms: General Rules

The inertia matrix, $\mathbf{M}(\mathbf{q})$, can be simplified by ordering the computations of the elements of $\mathbf{M}(\mathbf{q})$ so as to take advantage of as many common terms as possible: that is, develop a partial ordering of the computations using a recursive or hierarchical factoring scheme. The relationship between elements of $\mathbf{M}(\mathbf{q})$ can be seen by restructuring equation (9) using the relation $(\mathbf{a}\times\mathbf{b})\cdot(\mathbf{c}\times\mathbf{d}) = -\mathbf{a}\cdot\widehat{\mathbf{b}}\cdot\widehat{\mathbf{d}}\cdot\mathbf{c}$, where $\widehat{\mathbf{b}}$ is a second order tensor such that for any 3×1 vector $\mathbf{v}$, $\widehat{\mathbf{b}}\cdot\mathbf{v} = \mathbf{b}\times\mathbf{v}$ [Renaud]. With this relationship, and (4) and (5), the diagonal and off diagonal inertia matrix elements can be computed as:

$$M_{i,i} = \begin{cases} \mathbf{z}\cdot\left[\sum_{k=i}^{N} -m_k(\widehat{{}^i\mathbf{L}^*_{i,k}}\cdot\widehat{{}^i\mathbf{L}^*_{i,k}}) + I^{k^*/i}\right]\cdot\mathbf{z} \\ \quad = \mathbf{z}\cdot\mathbf{I}_{i,i}\cdot\mathbf{z} \quad \text{for i revolute} \\ \sum_{k=i}^{N} m_k(\mathbf{z}\cdot\mathbf{z}) = \mathbf{x}\cdot\left[\sum_{k=i}^{N} m_k(\hat{\mathbf{y}}\cdot\hat{\mathbf{y}})\right]\cdot\mathbf{x} \\ \quad = \mathbf{x}\cdot\mathbf{I}_{i,i}\cdot\mathbf{x} \quad \text{for i prismatic} \end{cases} \tag{17}$$

$$M_{i,j} = \begin{cases} {}^j\mathbf{z}_i\cdot\left[\sum_{k=j}^{N} -m_k(\widehat{{}^j\mathbf{L}^*_{i,k}}\cdot\widehat{{}^j\mathbf{L}^*_{j,k}}) + I^{k^*/j}\right]\cdot\mathbf{z} \\ \quad = {}^j\mathbf{z}_i\cdot\mathbf{I}_{i,j}\cdot\mathbf{z} \quad \text{for i \& j revolute} \\ {}^j\mathbf{x}_i\cdot\left[\sum_{k=j}^{N} -m_k(\widehat{{}^j\mathbf{y}_i}\cdot\widehat{{}^j\mathbf{L}^*_{j,k}})\right]\cdot\mathbf{z} \\ \quad = {}^j\mathbf{x}_i\cdot\mathbf{I}_{i,j}\cdot\mathbf{z} \quad \text{for i prismatic and j revolute} \\ {}^j\mathbf{z}_i\cdot\left[\sum_{k=j}^{N} -m_k(\widehat{{}^j\mathbf{L}^*_{i,k}}\cdot\hat{\mathbf{y}})\right]\cdot\mathbf{x} \\ \quad = {}^j\mathbf{z}_i\cdot\mathbf{I}_{i,j}\cdot\mathbf{x} \quad \text{for i revolute and j prismatic} \\ \sum_{k=j}^{N} -m_k{}^j\mathbf{x}_i\cdot\widehat{{}^j\mathbf{y}_i}\cdot\hat{\mathbf{y}}\cdot\mathbf{x} \\ \quad = {}^j\mathbf{x}_i\cdot\mathbf{I}_{i,j}\cdot\mathbf{x} \quad \text{for i \& j prismatic} \end{cases} \tag{18}$$

where $\mathbf{I}_{i,i}$ and $\mathbf{I}_{i,j}$ are the extended body inertia dyadics, ${}^j\mathbf{z}_i = {}^j\mathbf{R}^i\mathbf{z}$ is the unit vector $\mathbf{z}$ of C_i as expressed in C_j, and $I^{k^*/j}$ is the inertia dyad of link k as expressed in C_j.

Because of the serial nature of manipulators, many recursive relations can be derived between the extended body inertia dyadics. The identity ${}^i\mathbf{L}^*_{i,k} = {}^i\mathbf{L}^o_{i,j} + {}^i\mathbf{L}^*_{j,k}$ for $i < j < k$, can be used to derive the following relations for manipulators with revolute joints:

$$\begin{aligned} \mathbf{I}_{j,j} &= K_{j,j} - \{\mathbf{P}_{j+1}, {}^j\mathbf{U}_{j+1}\} + {}^j\mathbf{R}^{j+1}\mathbf{I}_{j+1,j+1}{}^{j+1}\mathbf{R}^j \\ \mathbf{I}_{j,j} &= K_{j,j} - \{\mathbf{P}_j, \mathbf{U}_{jo}\} - \widehat{{}^j\mathbf{U}_{j+1}}\cdot\widehat{\mathbf{P}_j} + {}^j\mathbf{R}^{j+1}\mathbf{I}_{j,j+1}{}^{j+1}\mathbf{R}^j \\ \mathbf{I}_{i,j} &= \mathbf{I}_{j,j} - \widehat{{}^j\mathbf{L}^o_{i,j}}\cdot\widehat{\mathbf{U}_j} \\ \mathbf{I}_{i,j} &= \mathbf{I}_{i+1,j} - \widehat{{}^j\mathbf{P}_{i+1}}\cdot\widehat{\mathbf{U}_j} \end{aligned} \tag{19}$$

where:

$$\begin{aligned} K_{j,j} &= I^{j^*/j} - m_j\widehat{r_j}\cdot\widehat{r_j} - \mathcal{M}_{j+1}(\widehat{\mathbf{P}_{j+1}}\cdot\widehat{\mathbf{P}_{j+1}}) \\ \mathcal{M}_j &= \sum_{k=j}^{N} m_k \\ \mathbf{U}_j &= \sum_{k=j}^{N} m_k{}^j\mathbf{L}^*_{j,k} = m_j\mathbf{r}_j + \mathcal{M}_{j+1}\mathbf{P}_j + {}^j\mathbf{U}_{j+1} \\ &= \mathbf{U}_{jo} + {}^j\mathbf{U}_{j+1} \end{aligned} \tag{20}$$

and the notation $\{\mathbf{a},\mathbf{b}\}$ is shorthand for $\widehat{\mathbf{a}}\cdot\widehat{\mathbf{b}} + \widehat{\mathbf{b}}\cdot\widehat{\mathbf{a}}$. Similary, for manipulators with both prismatic and revolute joints, the following relations can be derived (for joint i prismatic and joints j revolute):

$$\begin{aligned} \mathbf{I}_{i,j} &= {}^j\mathbf{R}^{j+1}\mathbf{I}_{i,j+1}{}^{j+1}\mathbf{R}^j - {}^j\hat{\mathbf{y}}_i\cdot\widehat{\mathbf{U}_{jo}} \\ \mathbf{I}_{j,i} &= \mathbf{I}_{j+1,i} - {}^i\widehat{\mathbf{U}_{io}}\cdot\hat{\mathbf{y}} \end{aligned} \tag{21}$$

These recursive identities can be used to minimize the amount of computation necessary to compute the inertia matrix terms, especially when the matrix elements are generated in symbolic form. The different recursion relations correspond to "diagonal", "across", and "down" recursion in the inertia matrix. The best method of recursion will depend on the particular manipulator geometry.

B. Simplification of Inertia Terms: Common Geometries

The formulation presented above is efficient for serial chain mechanisms of general geometry. However, almost all industrial manipulators have very regular geometries in which successive joints are either parallel or orthogonal. In addition, most six degree of freedom manipulators employ a "wrist" geometry in the last three joints. These regular geometries can lead to significant reductions in the complexity of the dynamics expressions. However, the factorizations in equations (19), (20), and (21) are not necessarily the most efficient for these special geometries. When special geometries arise, the dynamic matrix elements effected by these geometries are computed using a separate set of rules, and the remaining elements are computed using the more general algorithms of (19), (20), or (21).

For example, many manipulators contain two successive parallel revolute joints. The resulting simplifications can be determined by examining the partial velocities and partial angular velocities for parallel revolute joints. If joints p and $p+1$ are revolute and parallel, then:

$${}^p\mathbf{R}^{p+1}\mathbf{z} = {}^{p+1}\mathbf{R}^p\mathbf{z} = \mathbf{z} \tag{22}$$

and the relationship between partial velocities becomes:

$${}^j\tilde{\omega}^j_p = {}^j\tilde{\omega}^j_{p+1}; \qquad {}^p\tilde{V}^j_p = \mathbf{z}\times\mathbf{P}_{p+1} + {}^p\tilde{V}^j_{p+1}. \tag{23}$$

If we substitute (22), (23), and (19) into (17), then $M_{p,p+1}$ and $M_{p+1,p+1}$ can be related as:

$$\begin{aligned} M_{p,p+1} &= {}^{p+1}\mathbf{z}_p\cdot\left[\mathbf{I}_{p+1,p+1} - \widehat{\mathbf{P}_{p+1}}\cdot\widehat{\mathbf{U}_{p+1}}\right]\cdot\mathbf{z} \\ &= \mathbf{z}\cdot\mathbf{I}_{p+1,p+1}\cdot\mathbf{z} - \mathbf{z}\cdot{}^{p+1}\widehat{\mathbf{P}}_{p+1}\cdot\widehat{\mathbf{U}_{p+1}}\cdot\mathbf{z} \\ &= M_{p+1,p+1} + \eta_{p+1,p+1} \end{aligned} \tag{24}$$

where:

$$\eta_{i,j} = -{}^j\mathbf{z}_i\cdot\widehat{{}^j\mathbf{P}_i}\cdot\widehat{\mathbf{U}_j}\cdot\mathbf{z}$$

Using a similar derivation, the following other simplification rules for parallel joints can be developed:

$$\begin{aligned} M_{p,p} &= M_{p+1,p+1} + \mathbf{z}\cdot K_{p,p}\cdot\mathbf{z} + 2\eta_{p+1,p+1} \\ &= M_{p,p+1} + \mathbf{z}\cdot K_{p,p}\cdot\mathbf{z} + \eta_{p+1,p+1} \\ M_{i,p} &= M_{i,p+1} - {}^p\mathbf{z}_i\cdot\left[{}^p\widehat{\mathbf{U}_{p+1}}\cdot\widehat{\mathbf{P}_{p+1}} + \widehat{{}^p\mathbf{L}^o_{i,p}}\cdot\hat{\mathbf{U}}_{po}\right]\cdot\mathbf{z} \\ M_{p,i} &= M_{p+1,i} + \eta_{p+1,i} \end{aligned} \tag{25}$$

and $\mathbf{z}\cdot K_{p,p}\cdot\mathbf{z}$ is a constant independent of manipulator configuration, $\mathbf{q}$.

C. Simplification of Coriolis and Centripetal Terms: General Rules

The centripetal and coriolis terms can be simplified by considering some of the symmetries inherent in the Christoffel symbols. Because of the serial link nature of manipulators:

$$\frac{\partial M_{i,j}}{\partial q_k} = 0 \qquad \text{for } k \leq min(i,j) \tag{26}$$

Using this rule, a set of simplification rules can be established to reduce the number of coriolis and centripetal terms that need to be computed, regardless of manipulator geometry. For centripetal terms, the Christoffel symbol reduces to:

$$\mathbf{c}_{i,j,j} = \begin{cases} \partial M_{i,j}/\partial q_j & \text{for } i < j \\ 0 & \text{for } i = j \\ -\frac{1}{2}\partial M_{j,j}/\partial q_i & \text{for } i > j \end{cases} \tag{27}$$

Similar rules can be derived for the coriolis terms, $\mathbf{b}_{i,j,k}$. Using the fact that $k > j$ for the coriolis terms, the following general simplification rules for the coriolis terms can be derived from (15):

$$\mathbf{b}_{i,j,k} = \begin{cases} -2\,\mathbf{c}_{k,i,i} & \text{if } i = j \\ 0 & \text{if } i = k \\ -\mathbf{b}_{k,j,i} & \text{if } i > k \end{cases} \tag{28}$$

$$\mathbf{b}_{i,j,k} = \begin{cases} \partial M_{i,j}/\partial q_k + \partial M_{i,k}/\partial q_j & \text{if } i < j \\ \partial M_{i,j}/\partial q_k - \partial M_{j,k}/\partial q_i & \text{if } i > j \& i < k \end{cases} \tag{29}$$

D. Simplification of Coriolis and Centripetal Terms: Common Geometries

Because of equation (15), manipulator geometries which have a big impact on the structure of $\mathbf{M(q)}$ will have a corresponding impact on the structure of the centripetal and coriolis terms. To illustrate this point, the simplifications arising from parallel revolute joints are considered. As a representative derivation, $\mathbf{b}_{i,p,k}$ and $\mathbf{b}_{i,p+1,k}$ (for $i < p$) can be related as follows:

$$\mathbf{b}_{i,p,k} = \frac{\partial M_{i,p}}{\partial q_k} + \frac{\partial M_{i,k}}{\partial q_p} \tag{30}$$

Using equation (25) and Appendix I, $\partial M_{i,p}/\partial q_k$ can be found as

$$\frac{\partial M_{i,p}}{\partial q_k} = \frac{\partial M_{i,p+1}}{\partial q_k} - {}^{p}\mathbf{z}_i \cdot ({}^{p}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \mathbf{P}_{p+1} \cdot \mathbf{z} \tag{31}$$

Similarly, $\partial M_{i,k}/\partial q_p$ can be found as:

$$\begin{aligned} \partial M_{i,k}/\partial q_p = &-({}^{j}\mathbf{R}^{p+1}\hat{\mathbf{z}}^{p}\mathbf{z}_i) \cdot [\mathbf{I}_{j,j} - {}^{j}\widehat{\mathbf{L}^{o}_{i,j}} \cdot \widehat{\mathbf{U}_j}] \cdot \mathbf{z} \\ &+ {}^{j}\mathbf{z}_i \cdot ({}^{j}\widehat{\mathbf{R}^{p}\hat{\mathbf{z}}^{p}\mathbf{L}^{o}_{i,p}}) \cdot \widehat{\mathbf{U}_j} \cdot \mathbf{z} \end{aligned} \tag{32}$$

Using the fact that $\hat{\mathbf{z}}^{p}\mathbf{R}^{p+1} = {}^{p}\mathbf{R}^{p+1}\hat{\mathbf{z}}$, and the identity $(\mathbf{a} \times \mathbf{b}) \cdot (\mathbf{c} \times (\mathbf{d} \times \mathbf{e})) = -(\mathbf{d} \times \mathbf{e}) \cdot (\mathbf{c} \times (\mathbf{a} \times \mathbf{b}))$, the above equation can be rearranged to show that:

$$\frac{\partial M_{i,k}}{\partial q_p} = \frac{\partial M_{i,k}}{\partial q_{p+1}} + {}^{p}\mathbf{z}_i \cdot ({}^{p}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \widehat{\mathbf{P}_{p+1}} \cdot \mathbf{z} \tag{33}$$

Thus, for $i < p$:

$$\mathbf{b}_{i,p,k} = \mathbf{b}_{i,p+1,k} \tag{34}$$

It can also be shown that $\mathbf{b}_{i,p,k} = \mathbf{b}_{i,p+1,k}$ for $i > p+1$. Using similar derivations, the following additional relationships between the centripetal and coriolis matrix elements can be developed (which can be used when the simplification in equations (27) and (28) do not apply):

$$\mathbf{b}_{p+1,p,k} = \mathbf{b}_{p+1,p+1,k} - 2\,\mathbf{z} \cdot {}^{p}\widehat{\mathbf{P}_{p+1}} \cdot ({}^{p+1}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \mathbf{z}$$

$$\mathbf{b}_{p,i,p+1} = 0$$

$$\mathbf{b}_{i,p,p+1} = 2\,\mathbf{c}_{i,p+1,p+1} - {}^{p}\mathbf{z}_i \cdot ({}^{p}\widehat{\mathbf{R}^{p+1}\hat{\mathbf{z}}\mathbf{U}_{p+1}}) \cdot \widehat{\mathbf{P}_{p+1}} \cdot \mathbf{z}$$

$$\mathbf{b}_{p,k,j} = \mathbf{b}_{p+1,j,k} - 2^{j}\mathbf{z}_{p+1} \cdot {}^{j}\widehat{\mathbf{P}_{p+1}} \cdot ({}^{j}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \mathbf{z} \quad (p+1 < j)$$

$$\mathbf{b}_{p,j,k} = \mathbf{b}_{p+1,j,k} - 2\,{}^{p}\mathbf{z}_j \cdot ({}^{p}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \widehat{\mathbf{P}_{p+1}} \cdot \mathbf{z} \quad (j < p)$$

$$\mathbf{b}_{p,p+1,k} = -2\,\mathbf{c}_{k,p+1,p+1} - 2\,\mathbf{z} \cdot {}^{p+1}\widehat{\mathbf{P}_{p+1}} \cdot ({}^{p+1}\widehat{\mathbf{R}^{k}\hat{\mathbf{z}}\mathbf{U}_k}) \cdot \mathbf{z}$$

$$\mathbf{c}_{p,p+1,p+1} = -2\,\mathbf{c}_{p+1,p,p} + \mathbf{z} \cdot ({}^{p+1}\widehat{\mathbf{R}^{p}\hat{\mathbf{z}}\mathbf{P}_{p+1}}) \cdot \widehat{\mathbf{U}_{p+1}} \cdot \mathbf{z}$$

$$\mathbf{c}_{p,i,i} = \mathbf{c}_{p+1,i,i} - \mathbf{z} \cdot ({}^{i}\widehat{\mathbf{R}^{p}\hat{\mathbf{z}}\mathbf{P}_{p+1}}) \cdot \widehat{\mathbf{U}_i} \cdot \mathbf{z} \tag{35}$$

It is evident from the large number of relationships above that a substantial amount of simplification in the computation of the coriolis and centripetal terms can be obtained from special geometries.

5. EMDEG

The algorithm for generating manipulator dynamics and simplification procedures described above form the basis for a lisp-based program, EMDEG, that automatically generates the dynamic matrix and vector elements $M_{i,j}$,$\mathbf{c}_{i,j,j}$, $\mathbf{b}_{i,j,k}$, and g_i in symbolic form. For general geometries, EMDEG structures the computations according to (19), (20), or (21). However, when certain simplifying geometries arise, EMDEG will compute as many of the matrix elements as possible using the special simplification rules. EMDEG also includes other techniques for simplification including: trigonometric identities, rule-based hierarchical pattern matching, and factorization of expressions and subexpressions into configuration dependent and independent terms. The goal of these simplifications is to reduce the real time computation of the equations.

The simplification and pattern matching algorithms are based on a canonical sorting rule for dynamic expressions. After the initial derivation of $\mathbf{M(q)}$ using simplification rules, all matrix elements are converted to a sum of products form. Each term (denoted as a *SOPterm*) in the expression (denoted as a *SOPlist*) can be sorted to have the form:

constant × (*inertia term*) × (*trigonometric terms*)

Further, sorting rules can be used to canonically sort *SOPterms* in a *SOPlist*. The canonical arrangement of the dynamic expressions vastly simplifies the post-processing algorithm design.

After simplifying and factoring the inertia matrix terms into configuration dependent and independent terms, the centripetal, coriolis, and gravity terms are derived, with further simplification. Finally, iterative post-processing simplifications are applied to all of the dynamic expressions to find further common factors that can be factored out for further efficiencies. In addition to manipulator dynamic equations, EMDEG also generates the forward kinematic model, the manipulator jacobian matrix, and if desired, and analytic expression for the inverse of the jacobian matrix.

6. Example: the PUMA 560 Manipulator

To illustrate the use of this algorithm, the EMDEG program was applied to a PUMA 560 manipulator. The choice of coordinate frames is shown in figure 2, and the relevant kinematic and dynamic data is given below.

$${}^{0}\mathbf{R}^{1} = \begin{bmatrix} c_1 & -s_1 & 0 \\ s_1 & c_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad {}^{1}\mathbf{R}^{2} = \begin{bmatrix} c_2 & -s_2 & 0 \\ 0 & 0 & 1 \\ -s_2 & -c_2 & 0 \end{bmatrix};$$

$${}^{2}\mathbf{R}^{3} = \begin{bmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \quad {}^{3}\mathbf{R}^{4} = \begin{bmatrix} c_4 & -s_4 & 0 \\ 0 & 0 & -1 \\ s_4 & c_4 & 0 \end{bmatrix};$$

$${}^{4}\mathbf{R}^{5} = \begin{bmatrix} c_5 & -s_5 & 0 \\ 0 & 0 & 1 \\ -s_5 & -c_5 & 0 \end{bmatrix}; \quad {}^{5}\mathbf{R}^{6} = \begin{bmatrix} c_6 & -s_6 & 0 \\ 0 & 0 & -1 \\ s_6 & c_6 & 0 \end{bmatrix};$$

$$\mathbf{P}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad \mathbf{P}_2 = \begin{bmatrix} 0 \\ d_2 \\ 0 \end{bmatrix}; \quad \mathbf{P}_3 = \begin{bmatrix} a_2 \\ 0 \\ -d_3 \end{bmatrix}; \quad \mathbf{P}_4 = \begin{bmatrix} a_3 \\ -d_4 \\ 0 \end{bmatrix};$$

$$\mathbf{P}_5 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad \mathbf{P}_6 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad \mathbf{r}_1 = \begin{bmatrix} 0 \\ r_{y1} \\ 0 \end{bmatrix}; \quad \mathbf{r}_2 = \begin{bmatrix} r_{x2} \\ 0 \\ 0 \end{bmatrix};$$

$$\mathbf{r}_3 = \begin{bmatrix} 0 \\ r_{y3} \\ 0 \end{bmatrix}; \quad \mathbf{r}_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}; \quad \mathbf{r}_5 = \begin{bmatrix} 0 \\ r_{y5} \\ 0 \end{bmatrix}; \quad \mathbf{r}_6 = \begin{bmatrix} 0 \\ 0 \\ r_{z6} \end{bmatrix};$$

$$\mathbf{I}_1 = \begin{bmatrix} I_{xx1} & I_{xy1} & I_{xz1} \\ I_{xy1} & I_{yy1} & I_{yz1} \\ I_{xz1} & I_{yz1} & I_{zz1} \end{bmatrix}; \quad \mathbf{I}_2 = \begin{bmatrix} I_{xx2} & I_{xy2} & I_{xz2} \\ I_{xy2} & I_{yy2} & I_{yz2} \\ I_{xz2} & I_{yz2} & I_{zz2} \end{bmatrix};$$

$$\mathbf{I}_3 = \begin{bmatrix} I_{xx3} & I_{xy3} & I_{xz3} \\ I_{xy3} & I_{yy3} & I_{yz3} \\ I_{xz3} & I_{yz3} & I_{zz3} \end{bmatrix}; \quad \mathbf{I}_4 = \begin{bmatrix} I_{xx4} & 0 & 0 \\ 0 & I_{yy4} & 0 \\ 0 & 0 & I_{zz4} \end{bmatrix};$$

$$\mathbf{I}_5 = \begin{bmatrix} I_{xx5} & 0 & 0 \\ 0 & I_{xx5} & 0 \\ 0 & 0 & I_{zz5} \end{bmatrix}; \quad \mathbf{I}_6 = \begin{bmatrix} I_{xx6} & 0 & 0 \\ 0 & I_{xx6} & 0 \\ 0 & 0 & I_{zz6} \end{bmatrix};$$

The matrix and vector elements of the dynamic equations are given in Appendix II. The computation of the elements of $\mathbf{M}(\mathbf{q})$, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}^2)$, $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$, and $\mathbf{g}(\mathbf{q})$ require 291 multiplications and 131 additions (subtractions are counted the same as an additions). Using these results, the computation of the *inverse* dynamic problem requires an additional 110 multiplications and 123 additions, which results in a total computational burden of 401 multiplications and 254 additions, which is significantly less than the computational load of 852 multiplications and 738 additions as reported in in [Hollerbach]. EMDEG required approximately 10 minutes of CPU time on a VAX 11/780 to derive this solution.

Conclusion

The generation of manipulator dynamic equations in symbolic form is often necessary for many problems in manipulator design and control. This method of generating and simplifying the equations of motion, and the program EMDEG, which embodies further simplification procedures, has proven to be a useful tool for the symbolic generation of manipulator equations in a very efficient form.

Acknowledgements

The author gratefully acknowledges many helpful and stimulating discussions with Brian Armstrong, which led to improvements in the implementation of EMDEG. This research was made possible by National Science Foundation grant MEA80-19628, and the support of Professor Thomas Binford.

References

[Alden] M. J. Alden and A. Liegeois, "Computational Aspects in Robot Dynamics Modelling," Advanced Software in Robotics, ed. by A. Danthine and M. Geradin, Elsevier Science Publishers, North-Holland, 1984, pp. 3-14.

[Cesareo] G. Cesareo, F. Nicolo, and S. Nicosia, "DYMIR: A Code for Generating Dynamic Model of Robots," Advanced Software in Robotics, ed. by A. Danthine and M. Geradin, Elsevier Science Publishers, North-Holland, 1984, pp. 115-120.

[Craig] John J. Craig, Introduction to Robotics: Manipulation and Control, Addison-Wesley, Reading, Mass, 1986.

[Hollerbach] John M. Hollerbach, "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity", *I.E.E.E. Transactions on Systems, Man, and Cybernetics*, vol. SMC-10, no. 11, pp. 730-736, November 1980.

[Kane] Thomas R. Kane and David A. Levinson, "The Use of Kane's Dynamical Equations in Robotics", *The International Journal of Robotics Research*, vol. 2, No. 3, Fall 1983, pp. 3-20.

[Khatib and Burdick] Oussama Khatib and Joel W. Burdick, "Dynamic Optimization in Manipulator Design: the Operational Space Approach", ASME Annual Winter Meeting, Miami Fla, November, 1985.

[Luh,Walker,Paul] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "On-Line Computational Scheme for Mechanical Manipulators," *ASME J. of Dynamic Systems, Measurement, and Control*, vol. 102, pp. 69-76, June, 1980.

[Murray] John J. Murray and Charles P. Neuman, "Arm: an Algebraic Robot Dynamic Modeling Program", *Proc. 1984 International Conference on Robotics*, Atlanta, Georgia, March 13-15, 1984, pp. 115-120.

[Megahed] S. Megahed and M. Renaud, "Minimization of the Computation Time Necessary for the Dynamic Control of Robot Manipulators", *Proc. of the 12th International Symposium on Industrial Robots/6th Conference on Industrial Robot Technology*, Paris, June 9-11, 1982, pp. 469-478.

[Paul] Richard P. Paul, Robot Manipulators: Mathematics, Programming, and Control, MIT Press, Cambridge Mass., 1981.

[Renaud 84] Marc Renaud, "An Efficient Iterative Analytic Procedure for Obtaining a Robot Manipulator Dynamic Model," Robotics Research, ed. by M. Brady and R. Paul, MIT Press, Cambridge, 1984, pp. 749-764.

Appendix I: Some Useful Derivatives

The following derivations are for revolute jointed manipulators. Analogous expressions for manipulators with mixtures of prismatic and revolute joints can be derived.

1. Derivatives of Rotation matrices

The derivative of a general rotation matrix, ${}^i\mathbf{R}^j$, with respect to joint p can be found by first considering the derivative of ${}^{p-1}\mathbf{R}^p$ with respect to joint p:

$$\frac{\partial\left({}^{p-1}\mathbf{R}^p\right)}{\partial q_p} = {}^{p-1}\mathbf{R}^p\, \tilde{\mathbf{z}} \qquad (A-1)$$

where $\tilde{\mathbf{z}}$ is a 3×3 matrix such that for any 3×1 vector v, $\tilde{\mathbf{z}}v = \mathbf{z}\times v$. Hence, the more general rotation matrix derivative is :

$$\frac{\partial\left({}^i\mathbf{R}^j\right)}{\partial q_p} = \frac{\partial\left({}^i\mathbf{R}^{p-1}\,{}^{p-1}\mathbf{R}^p\,{}^p\mathbf{R}^j\right)}{\partial q_p} = {}^i\mathbf{R}^{p-1}\frac{\partial\left({}^{p-1}\mathbf{R}^p\right)}{\partial q_p}{}^p\mathbf{R}^j \qquad (A-2)$$
$$= {}^i\mathbf{R}^{p-1}\,{}^{p-1}\mathbf{R}^p\,\tilde{\mathbf{z}}\,{}^p\mathbf{R}^j = {}^i\mathbf{R}^p\,\tilde{\mathbf{z}}\,{}^p\mathbf{R}^j$$

for $i < p \le j$, else $\partial\,{}^i\mathbf{R}^j/\partial q_p = 0$.

2. Derivative of ${}^i\mathbf{L}^*_{i,k}$:

Starting from equation (6), and using the identity, $\partial\,{}^i\mathbf{R}^m/\partial q_p = 0$ for $m < p$:

$$\frac{\partial\,{}^i\mathbf{L}^*_{i,k}}{\partial q_p} = \frac{\partial}{\partial q_p}\left[\sum_{m=i}^{p-1}{}^i\mathbf{R}^m\,\mathbf{P}_{m+1} + \sum_{m=p}^{j-1}{}^i\mathbf{R}^m\,\mathbf{P}_{m+1} + {}^i\mathbf{R}^k\,\mathbf{r}^*_k\right]$$
$$= \sum_{m=p}^{j-1}\frac{\partial\,{}^i\mathbf{R}^m}{\partial q_p}\mathbf{P}_{m+1} + \frac{\partial\,{}^i\mathbf{R}^k}{\partial q_p}\mathbf{r}^*_k$$
$$= \sum_{m=p}^{j-1}{}^i\mathbf{R}^p\,\tilde{\mathbf{z}}\,\mathbf{P}_{m+1} + {}^i\mathbf{R}^p\,\tilde{\mathbf{z}}\,{}^p\mathbf{R}^k\,\mathbf{r}^*_k = {}^i\mathbf{R}^p\tilde{\mathbf{z}}^p\mathbf{L}^*_{p,k}) \qquad (A-3)$$

3. Derivative of $\mathbf{U}_j$:

$$\begin{aligned} \frac{\partial \mathbf{U}_j}{\partial q_k} &= \frac{\partial}{\partial q_k}\left[\sum_{m=j}^{N} m_m {}^j\mathbf{L}^*_{j,m}\right] \\ &= \sum_{m=k}^{N} m_m \, {}^j\mathbf{R}^k \, \hat{\mathbf{z}} \, {}^k\mathbf{L}^*_{k,m} = {}^j\mathbf{R}^k \, \hat{\mathbf{z}} \, \mathbf{U}_k \end{aligned} \qquad (A-4)$$

Appendix II: Dynamic Equations for the PUMA 560

In the following equations, s_i, c_i denotes $sin(\theta_i), cos(\theta_i)$. The inertia matrix elements are:

$$\begin{aligned}
M_{11} &= K_{11} - K_{111}T_1 + 2I_{yz_2}T_3 + 2V_5T_6 - V_6T_4 + V_8T_{21} \\
&\quad + 2V_7T_{22} + K_{112}T_{10} - K_{11102}T_{14} - K_{11132}T_{13} \\
M_{12} &= M_{13} - K_{1115}s_2T_{13} + K_{121}s_2 - I_{yz_2}c_2 \\
M_{13} &= +V_9c_{23}T_{13} - V_1s_{23}T_9 - V_{10}s_{23}s_5 + K_{1113}c_{23}c_5 + K_{123}s_{23} \\
&\quad - K_{125}c_{23} \\
M_{14} &= V_2T_{17} - V_4T_{14} - K_{1113}c_{23}T_{13} + M_{44}c_{23} \\
M_{15} &= -V_4T_{15} + K_{1113}c_{23}T_{16} - K_{1113}s_{23}s_5 + V_3T_{18} \\
M_{16} &= I_{zz6}c_{23}c_5 - I_{zz6}s_{23}T_{14} \\
M_{22} &= K_{22} + M_{23} + \eta \\
M_{23} &= M_{33} + \eta \\
M_{24} &= M_{34} + K_{1115}s_3T_{13} \\
M_{25} &= M_{35} - K_{1115}s_3T_{16} - K_{1115}c_3s_5 \\
M_{26} &= M_{36} \\
M_{33} &= K_{33} + V_1T_7 - K_{11102}T_{14} - K_{11162}c_5 \\
M_{34} &= -V_2s_4 \\
M_{35} &= -K_{1110}s_5 + V_3c_4 \\
M_{36} &= I_{zz6}T_{13} \\
M_{44} &= K_{44} + K_{112}T_{10} \\
M_{45} &= 0 \qquad M_{46} = I_{zz6}c_5 \\
M_{55} &= K_{55} \qquad M_{56} = 0 \qquad M_{66} = K_{66}
\end{aligned}$$

The non-zero centripetal terms are:

$$\begin{aligned}
c_{122} &= c_{133} + I_{yz_2}s_2 + K_{121}c_2 - K_{1115}c_2T_{13} \\
c_{133} &= -V_9s_{23}T_{13} + K_{125}s_{23} - V_1c_{23}T_9 + K_{123}c_{23} - V_{10}T_{20} \\
&\quad - K_{1113}T_{19} \\
c_{144} &= -K_{112}T_{18}T_{12} + V_{12} + V_{11}T_{20} \\
c_{155} &= +V_{11}T_{20} + V_{12} - K_{1113}T_{19} \\
c_{211} &= c_{311} + K_{111}T_3 + I_{zy_2}(T_1 - T_2) - K_{1115}s_2c_{23}T_{14} + V_7T_{13} \\
&\quad + K_{1111}T_{21} \\
c_{233} &= V_8c_3 - V_7s_3 \\
c_{244} &= -V_2c_4 + K_{1115}s_3T_{14} \\
c_{255} &= c_{355} + K_{1115}s_3T_{14} - K_{1115}c_3c_5 \\
c_{311} &= V_5(T_4 - T_5) + V_6T_6 + V_7T_{21} - V_8T_{22} \\
c_{322} &= -c_{233} \\
c_{344} &= V_9T_{14} \\
c_{355} &= -K_{1110}c_5 + K_{1116}T_{14} \\
c_{411} &= V_2T_6s_4 - K_{1115}T_{22}T_{13} - V_{11}s_5 + V_{13}T_4s_4 \\
c_{422} &= c_{433} - K_{1115}c_3T_{13} \\
c_{433} &= -V_{13}s_4 \\
c_{511} &= +(-K_{1116}s_5 - K_{112}T_7T_{12} + K_{1122}T_{12} - K_{1110}T_{16})T_4 \\
&\quad - DV_{55}T_6 - K_{112}T_{12} + K_{1115}T_{22}T_{16} - K_{1115}T_{21}s_5 + V_{10}c_5 \\
c_{522} &= c_{533} + K_{1115}c_3T_{16} - K_{1115}s_3s_5 \\
c_{533} &= V_{14}s_5 + K_{1110}T_{16} \\
c_{544} &= -K_{112}T_{12}
\end{aligned}$$

The non-zero coriolis parameters are:

$$\begin{aligned}
&b_{112} = -2c_{211} \quad b_{113} = -2c_{311} \quad b_{114} = -2c_{411} \\
&b_{115} = -2c_{511} \quad b_{123} = 2c_{133} \quad b_{124} = b_{134} \\
&b_{125} = b_{135} \quad b_{126} = b_{136} \quad b_{134} = D_{134} + D_{143} \\
&b_{135} = D_{135} + D_{153} \quad b_{136} = -b_{316} \\
&b_{145} = D_{145} + D_{154} \quad b_{146} = I_{zz6}s_{23}T_{13} \\
&b_{156} = -b_{516} \quad b_{214} = b_{314} - K_{11152}s_2T_{14} \\
&b_{215} = b_{315} - K_{11152}s_2T_{15} \quad b_{216} = b_{316} \quad b_{223} = -2c_{322} \\
&b_{224} = -2c_{422} \quad b_{225} = -2c_{522} \quad b_{234} = b_{334} \\
&b_{235} = b_{335} - K_{11152}c_3T_{16} + K_{11152}s_3s_5 \\
&b_{245} = b_{345} + K_{11152}s_3T_{15} \quad b_{246} = b_{346} \\
&b_{256} = b_{356} \quad b_{312} = -b_{213} \quad b_{314} = D_{134} - D_{143} \\
&b_{315} = D_{135} - D_{153} \quad b_{316} = I_{zz6}T_{19} + I_{zz6}c_{23}T_{14} \\
&b_{324} = b_{334} \quad b_{325} = b_{335} \quad b_{334} = -2c_{433} \\
&b_{335} = -2c_{533} \quad b_{345} = D_{345} + D_{354} \quad b_{346} = -b_{436} \\
&b_{356} = -b_{536} \quad b_{412} = -b_{214} \quad b_{413} = -b_{314} \\
&b_{415} = D_{145} - D_{154} \quad b_{416} = -I_{zz6}s_{23}T_{13} \\
&b_{423} = -b_{324} \quad b_{425} = b_{435} \quad b_{426} = b_{436} \\
&b_{435} = D_{345} - D_{354} \quad b_{436} = -I_{zz6}T_{14} \\
&b_{445} = -2c_{544} \quad b_{456} = -b_{546} \quad b_{512} = -b_{215} \\
&b_{513} = -b_{315} \quad b_{514} = -b_{415} \\
&b_{516} = I_{zz6}T_{20} + I_{zz6}s_{23}T_{16} \quad b_{523} = -b_{325} \\
&b_{524} = -b_{425} \quad b_{526} = b_{536} \quad b_{534} = -b_{435} \\
&b_{536} = -I_{zz6}T_{15} \quad b_{546} = I_{zz6}s_5 \quad b_{612} = -b_{216} \\
&b_{613} = -b_{316} \quad b_{614} = -b_{416} \quad b_{615} = -b_{516} \\
&b_{623} = -b_{326} \quad b_{624} = -b_{426} \quad b_{625} = -b_{526} \\
&b_{634} = -b_{436} \quad b_{635} = -b_{536} \quad b_{645} = -b_{546}
\end{aligned}$$

The elements of the gravity vector are

$$\begin{aligned}
g_1 &= 0 \\
g_2 &= g_{211}c2 + g_3 \\
g_3 &= g_{311}s_{23} + g_{312}c_{23} + g_{313}c_{23}T_{14} + g_{313}T_{19} \\
g_4 &= -g_{313}s_{23}T_{13} \\
g_5 &= g_{511}s_{23}T_{16} + g_{311}c_{23}s5 \\
g_6 &= 0
\end{aligned}$$

The following terms are configuration dependent, and must be computed in real time:

$$\begin{aligned}
&T_1 = s_2s_2 \quad T_2 = c_2c_2 \quad T_3 = s_2c_2 \quad T_4 = s_{23}s_{23} \\
&T_5 = c_{23}c_{23} \quad T_6 = s_{23}c_{23} \quad T_7 = s_4s_4 \quad T_8 = c_4c_4 \\
&T_9 = s_4c_4 \quad T_{10} = s_5s_5 \quad T_{11} = c_5c_5 \quad T_{12} = s_5c_5 \\
&T_{13} = s_4s_5 \quad T_{14} = c_4s_5 \quad T_{15} = s_4c_5 \quad T_{16} = c_4c_5 \\
&T_{17} = s_{23}c_4 \quad T_{18} = s_{23}s_4 \quad T_{19} = s_{23}c_5 \\
&T_{20} = c_{23}s_5 \quad T_{21} = c_2s_{23} \quad T_{22} = c_2c_{23}
\end{aligned}$$

$$\begin{aligned}
\eta &= V_8s_3 + V_7c_3 \\
V_1 &= K_{119} - K_{112}T_{10} \qquad V_2 = K_{112}T_{12} - K_{1116}s_5 \\
V_3 &= M_{55} - K_{1116}c_5 \qquad V_4 = K_{1110}c_{73} + K_{1115}c_2 \\
V_5 &= V_2c_4 + K_{115} - K_{1110}c_5 \\
V_6 &= K_{1122}T_{10} + K_{11162}c_5 - K_{112}T_7T_{10} - K_{113} \\
&\quad - K_{11102}T_{14} + K_{119}T_7 \\
V_7 &= K_{1111} - K_{1115}T_{14} \qquad V_8 = K_{117} - K_{1115}c_5
\end{aligned}$$

$$V_9 = K_{1116} - K_{112}c_5 \qquad V_{10} = K_{1110}s_4 + K_{1113}c_4$$
$$V_{11} = K_{1110}s_4 - K_{1113}c_4 \qquad V_{12} = (K_{1116}s_{23} + K_{1115}c_2)T_{13}$$
$$V_{13} = V_1c_4 + K_{1110}s_5 \qquad V_{14} = K_{112}T_7c_5 - K_{1116}$$
$$DV_{25} = K_{112}(T_{11} - T_{10}) - K_{1116}c_5 \qquad DV_{55} = DV_{25}c_4 + K_{1110}s_5$$
$$D_{134} = +V_9c_{23}T_{14} - V_1s_{23}T_8 + V_1s_{23}T_7$$
$$D_{135} = V_8c_{23}T_{15} - V_{10}T_{19} - K_{1113}T_{20}$$
$$D_{143} = V_2c_{23}c_4 + K_{1110}s_{23}T_{14} + K_{1113}s_{23}T_{13} - M_{44}s_{23}$$
$$D_{145} = DV_{25}T_{17} - K_{1110}c_4T_{10} - K_{1113}c_{23}T_{13}$$
$$D_{153} = -K_{1113}s_{23}T_{16} - K_{1113}T_{20} + V_3c_{23}s_4$$
$$D_{154} = -V_4T_{16} - K_{1113}c_{23}T_{15} + V_3T_{17} \qquad D_{345} = -DV_{25}s_5$$
$$D_{354} = -V_3s_4$$

These terms are configuration independent and can be computed once, and then stored as constants:

$$\begin{aligned}
K_{111} &= -I_{xx2} + I_{yy_2} + m_2r_{x2}r_{x2} + m_3a_2a_2 + m_4a_2a_2 \\
&\quad + m_5a_2a_2 + m_6a_2a_2 \\
K_{112} &= m_5r_{y_5}r_{y_5} + I_{xx6} - I_{zz6} + m_6r_{z6}r_{z6} \\
K_{113} &= I_{xx3} - I_{yy_3} + m_3r_{y_3}r_{y_3} + I_{xx4} - I_{zz4} \\
&\quad + m_4d_4d_4 - m_4a_3a_3 + m_5d_4d_4 - m_5a_3a_3 + m_5r_{y_5}r_{y_5} \\
&\quad + I_{xx6} - I_{zz6} + m_6d_4d_4 - m_6a_3a_3 + m_6r_{z6}r_{z6} \\
K_{115} &= I_{xy_3} + m_4a_3d_4 + m_5a_3d_4 + m_6a_3d_4 \\
K_{117} &= -m_3a_2r_{y_3} + m_4a_2d_4 + m_5a_2d_4 + m_6a_2d_4 \\
K_{119} &= I_{xx4} - I_{yy_4} + I_{xx5} - I_{zz5} \\
K_{1110} &= m_5a_3r_{y_5} - m_6a_3r_{z6} \\
K_{1111} &= m_4a_2a_3 + m_5a_2a_3 + m_6a_2a_3 \\
K_{1113} &= m_5d_2r_{y_5} + m_5d_3r_{y_5} - m_6d_2r_{z6} - m_6d_3r_{z6} \\
K_{1115} &= m_5a_2r_{y_5} - m_6a_2r_{z6} \\
K_{1116} &= m_5d_4r_{y_5} - m_6d_4r_{z6} \\
K_{121} &= -I_{xz2} + m_2d_2r_{x2} + m_3a_2d_2 + m_3a_2d_3 + m_4a_2d_2 \\
&\quad + m_4a_2d_3 + m_5a_2d_2 + m_5a_2d_3 + m_6a_2d_2 + m_6a_2d_3 \\
K_{123} &= -I_{xz3} + m_4a_3d_2 + m_4a_3d_3 + m_5a_3d_2 + m_5a_3d_3 \\
&\quad + m_6a_3d_2 + m_6a_3d_3 \\
K_{125} &= I_{yz_3} - m_3d_2r_{y_3} - m_3d_3r_{y_3} + m_4d_2d_4 + m_4d_3d_4 \\
&\quad + m_5d_2d_4 + m_5d_3d_4 + m_6d_2d_4 + m_6d_3d_4 \\
K_{141} &= I_{zz4} + I_{xx5} + I_{zz6} \\
K_{151} &= I_{zz5} + m_5r_{y_5}r_{y_5} + I_{xx6} + m_6r_{z6}r_{z6}
\end{aligned}$$

$$K_{1122} = 2K_{112} \qquad K_{11102} = 2K_{1110} \qquad K_{11162} = 2K_{1116}$$
$$K_{11132} = 2K_{1113} \qquad K_{11152} = 2K_{1115}$$

$$\begin{aligned}
g_{211} &= m_2r_{x2} + m_3a_2 + m_5a_2 + m_6a_2 + m_4a_2 \\
g_{311} &= m_3r_{y_3} + m_4d_4 + m_5d_4 + m_6d_4 \\
g_{312} &= m_4a_3 + m_5a_3 + m_6a_3 \\
g_{313} &= m_6r_{z6} - m_5r_{y_5} \\
g_{511} &= m_6r_{z6} + m_5r_{y_5}
\end{aligned}$$

$$\begin{aligned}
K_{11} &= m_5a_3a_3 + m_5d_3d_3 + 2m_5d_2d_3 + m_5d_2d_2 + I_{xx5} + m_5a_2a_2 \\
&\quad + m_6a_2a_2 + m_6d_2d_2 + 2m_6d_2d_3 + I_{zz_4} + m_6d_3d_3 + m_6a_3a_3 \\
&\quad + m_4a_3a_3 + m_4d_3d_3 + 2m_4d_2d_3 + m_4d_2d_2 + m_4a_2a_2 + I_{yy_3} \\
&\quad + m_3d_3d_3 + 2m_3d_2d_3 + m_3d_2d_2 + m_3a_2a_2 + I_{yy_2} + m_2d_2d_2 \\
&\quad + I_{zz6} + m_2r_{x2}r_{x2} + m_1r_{y_1}r_{y_1} + I_{zz1} \\
K_{22} &= m_6a_2a_2 + m_6d_4d_4 + I_{zz5} + m_6a_3a_3 + m_5r_{y_5}r_{y_5} \\
&\quad + m_5a_3a_3 + m_6r_{z6}r_{z6} + m_5d_4d_4 + m_5a_2a_2 + I_{xx6} \\
&\quad + I_{yy_4} + m_4d_4d_4 + m_4a_3a_3 + m_4a_2a_2 + m_3r_{y_3}r_{y_3} \\
&\quad + m_3a_2a_2 + I_{zz3} + m_2r_{x2}r_{x2} + I_{zz2} \\
K_{23} &= I_{zz5} + m_6a_3a_3 + m_6d_4d_4 + m_5r_{y_5}r_{y_5} + m_5d_4d_4 \\
&\quad + m_5a_3a_3 + m_6r_{z6}r_{z6} + I_{xx6} + I_{yy_4} + m_4d_4d_4 \\
&\quad + m_4a_3a_3 + m_3r_{y_3}r_{y_3} + I_{zz3}
\end{aligned}$$

$$K_{33} = K_{23} \qquad K_{44} = I_{zz6} + I_{xx5} + I_{zz4}$$
$$K_{55} = I_{xx6} + m_6r_{z6}r_{z6} + m_5r_{y_5}r_{y_5} + I_{zz5} \qquad K_{66} = I_{zz6}$$

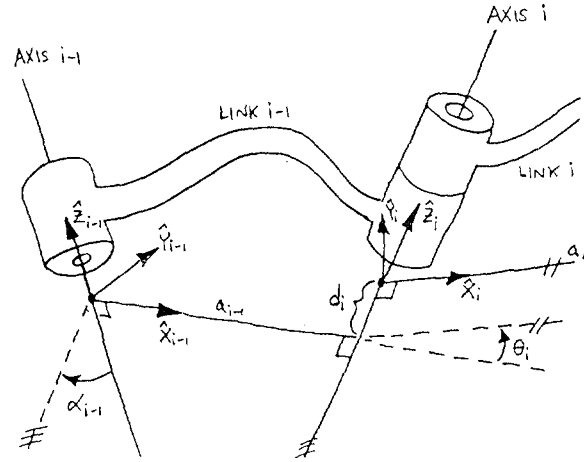

Figure 1. Kinematic Description of a Link [Craig]

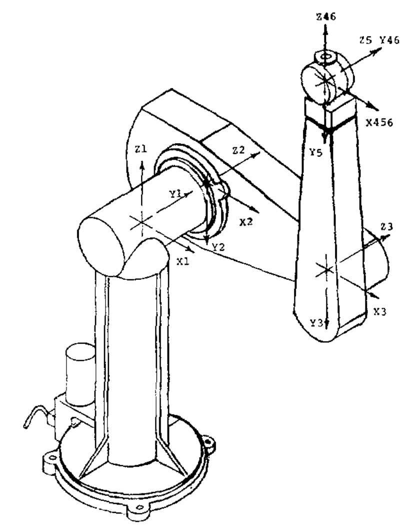

Figure 2. PUMA 560 Manipulator